Underlying Representation in Optimality Theory*

Sunghoon Hong

This paper addresses how we should deal with underlying representation (UR) in Optimality Theory (OT), focusing in particular on the proper method of Lexicon Optimization (LO), a mechanism in OT that determines a 'real' input among various possible inputs. After closely examining previous proposals, I conclude that Kager's (1999) model of LO is most promising, only if it is controlled by a meta-principle, Input-Output Isomorphy, that guarantees a UR to be identical to its surface form in the cases where underlying contrasts are expressed at surface level. I demonstrate that LO performed in this way provides a good account of UR for allomorphy cases without posing any problem to Richness of the Base, one of the fundamental concepts in OT.

1. Introduction

Optimality Theory (OT) heavily makes use of output-based constraints, and the role of underlying representation (UR) in this theory thus becomes only marginal. The burden is shifted entirely on constraints and the ranking among them, and in principle, no restrictions on UR or input are imposed. Theoretically, this is ensured by Richness of the Base, one of the fundamental tenets of OT.

(1) Richness of the Base (ROB) (Tesar/Smolensky 1998 : 252)
The set of possible inputs to the grammar of all languages is the same. The grammatical inventories of languages are defined as the

* This research was financially supported by Hansung University in the year of 2000. I would like to express my gratitude to two anonymous reviewer for their valuable comments. The paper is a part of the ongoing project of the author that deals with the underlying representation and Lexicon Optimization in Optimality Theory. A further continuation of this research is found in Hong (2001).

forms appearing in the structural descriptions that emerge from the grammar when it is fed by the universal set of all possible inputs.

According to ROB, the set of possible inputs to the grammar are the same across languages, and by this proposition, ROB attributes all systematic cross-linguistic variation entirely to constraint ranking (Prince/Smolensky 1993, Smolensky 1996, Tesar/Smolensky 1996, 1998, 2000). When cast into an actual analysis, this means that a correct output is guaranteed without hinging on any particular input, if provided with proper constraint ranking.

It was noted, however, that ROB might pose a serious computation problem on learnability. Since any input must converge on a specific output at hand, learning a grammar, whose basic function is to map an input to an output, may become a burden on the part of a learner. From the viewpoint of learnability, therefore, input or UR must be fixed for a given output. Lexicon Optimization (LO), the basic idea of which is that a proper UR automatically follows from harmonic evaluation germane to OT, comes into play to serve this purpose.

Other than this basic definition, however, many aspects of LO still remain unclarified. Among other things, it needs to be brought into light how LO is actually implemented. We need to settle, in particular, how LO applies to alternating or allomorphy cases.

The purpose of this paper is to explore the proper model of LO and examine how we should treat the UR in OT. To this end, we will introduce in section 2 the original formulation of LO, proposed by Prince/Smolensky (1993), and point out that LO is relevant only when multiple inputs converge on a single output form. Centering on voicing alternation in Dutch, we will show in section 3 that Prince/Smolensky's formulation of LO encounters a problem when it applies to alternating or allomorphy cases, as first noted by Inkelas (1995). In section 4, we will examine Inkelas (1995) and Tesar/Smolensky (1996, 2000), both of which propose a version of LO that attempts to cover allomorphy cases, and point out that they share a fundamental drawback. In section 5, we will discuss Kager's (1999) model and show that his version of LO is most promising, provided that some proper addenda are added. Then, in section 6, we will demonstrate how the typological variants of Dutch obtain an explanation in this model. Finally, section 7 will serve as a summary and conclusion of this paper.

2. Contrast, Neutralization, and Lexicon Optimization

As we have mentioned, no specific input is required for an output due to ROB, and thus there are potentially multiple inputs for a single output. The basic idea of LO, defined by Prince/Smolensky (1993 : 192) as below, is that we can readily track down the "optimal" inputs ("optimal" in terms of learnability) among those potential inputs, given proper constraint ranking.

(2) Lexicon Optimization (LO)
"Suppose that several different inputs $I_1, I_2, ..., I_n$, when parsed by a grammar G lead to corresponding outputs $O_1, O_2, ..., O_n$, all of which are realized as the same phonetic form ϕ--these inputs are all *phonetically equivalent* with respect to G. Now one of these outputs must be the most harmonic, by virtue of incurring the least significant violation marks: suppose this optimal one is labelled O_k. Then the learner should choose, as the underlying form for ϕ, the input I_k."

According to this definition of LO, the real input to an output is the form whose mapping to the output involves the least significant constraint violations, among all potential inputs whose outputs all converge on the same phonetic form.

To illustrate how LO works, let us consider the following hypothetical case where the inputs, /CV/, /CVV/, and /C/, all converge on a phonetic form, [CV], via the outputs, CV, CV<V>, and C□, respectively[1].

(3)

Inputs		Outputs		Surface Phonetic Form
/CV/	→	CV	→	
/CVV/	→	CV<V>	→	CV
/C/	→	C□	→	

The optimal input for the phonetic form [CV] is determined by the harmonic evaluation of the three input-output mapping relations, represented here as $M_1(CV)=CV$, $M_2(CVV)=CV\langle V\rangle$, and $M_3(C)=C\square$. In the first mapping, the input maps to the output that is identical, observing the

[1] In conformity with Prince/Smolensky's (1993) original version of LO formulated in the Containment model, we make use of output forms such as CV<V> and C□, both of which are phonetically realized as [CV].

faithfulness constraints; in the second and third mapping, a faithfulness constraint, Parse or Fill, is violated. Of notice is that all the inputs realize into a single phonetic form, which suggests that there is a tie on markedness constraints. The deciding factor, therefore, is the faithfulness constraints, and the first mapping, which does not violate any faithfulness constraints, is evaluated as the most harmonic, and /CV/ is selected as the UR.

Let us now turn to an actual example, vowel nasalization. In OT terms, vowel nasalization can be characterized by the interaction of the markedness and faithfulness constraints as below, of which the former is ranked higher than the latter.

(4) a. $*V_{oral}N$:
Vowels must not be oral before a nasal consonant.

b. Ident-IO(nasal):
Corresponding segments in input and output have identical values for [nasal]. (faithfulness)

Importantly, given the constraints and the ranking as above, vowel nasalization here is guaranteed without hinging on a particular input, which is exactly predicted by ROB. As we see in the following tableaux, the optimal output is chosen regardless of its input being /pæn/ or /pæ̃n/.

(5) Markedness ≫ Faithfulness

input: /pæn/	$*V_{oral}N$	Ident-IO(nasal)
[pæn]	*!	
☞ [pæ̃n]		*

input: /pæ̃n/	$*V_{oral}N$	Ident-IO(nasal)
[pæn]	*!	*
☞ [pæ̃n]		

LO comes into play at this point to determine the real input. As we pointed out above, inputs must be fixed to ease the unnecessary computation burden on the part of grammar learners. This is made possible by LO, which evaluates each input-output mapping that produces the optimal output form. In the case above, LO compares /pæn/→[pæ̃n] and /pæ̃n/→[pæ̃n], and selects the latter as optimal, which has a less degree of violation. Accordingly, /pæ̃n/ is determined as the optimal input for [pæ̃n].

(6) Lexicon Optimization

inputs		outputs	$*V_{oral}N$	Ident-IO(nasal)
/pæn/	→	[pæ̃n]		*!
☞ /pæ̃n/	→	[pæ̃n]		

It should be noted that LO is important only when multiple inputs converge on the same output form ([pæ̃n] in the case at hand). Logically, such cases of neutralization takes place only if markedness constraints are higher-ranked than faithfulness constraints. If the ranking is the other way around (i.e. faithfulness outranks markedness), each input form is mapped to its own output, which is identical to the input, and hence, different inputs do not converge on the same output form. In such cases, LO is not necessary to determine an input form, and inputs are simply those equal to the output forms.

(7) Faithfulness ≫ Markedness

input: /pæn/	Ident-IO(nasal)	$*V_{oral}N$
☞ [pæn]		*
[pæ̃n]	*!	

input: /pæ̃n/	Ident-IO(nasal)	$*V_{oral}N$
[pæn]	*!	*
☞ [pæ̃n]		

In summary, the importance of LO reveals only in the cases where multiple inputs are neutralized into a single output form due to high ranking of markedness. These are the cases of "contrast neutralized," where LO selects the input form whose mapping to its output incurs least violations (see (8a)). Where faithfulness dominates markedness, on the other hand, inputs are simply mapped to the forms that are identical to them, and these are referred to as the cases of "contrast expressed" (see (8b)).

(8) a. Different inputs map to a single output (contrast neutralized):
Markedness ≫ Faithfulness

inputs		output				real inputs
/pæn/	→	[pæ̃n]	——	LO	→	/pæ̃n/
/pæ̃n/	↗					

b. Different inputs map to their own outputs (contrast expressed): Faithfulness ≫ Markedness

inputs		output		real inputs
/pæn/	⟶	[pæn]	⇢	/pæn/
/pæ̃n/	⟶	[pæ̃n]	⇢	/pæ̃n/

3. Allomorphy: A Challenge to Lexicon Optimization

In the previous section, we saw that the original formulation of LO is mainly concerned with the cases where inputs converge on a single phonetic output form. These are the cases where there are no alternations involved. In this section, we will examine how LO fares in dealing with alternations.

The case we will look into is Dutch voicing alternation, of which the representative examples are given below (cf. Kager 1999).

(9)	singular	plural	
a.	[pɛt]	[pɛ.tən]	'cap'
b.	[bɛt]	[bɛ.dən]	'bed'

The examples in (9a) show no alternation, but in (9b) there is alternation between [bɛt] and [bɛd]. Adopting Lombardi's (1999) analysis of syllable-final devoicing, we can account for such voicing alternation with the three constraints below.

(10) a. IdentOnset(voice)
Consonants in the syllable onset position should be faithful to underlying [voice] specification.

b. *Voice
Do not have [voice] feature.

c. Ident-IO(voice)
Consonants should be faithful to underlying [voice] specification.

Let us examine in detail how the examples in (9) are produced. First of all, the singular examples [pɛt] and [bɛt], in which the final coda obstruents always appear voiceless, are produced by the ranked consonants *Voice ≫ Ident-IO(voice). On this ranking, [pɛt] and [bɛt] are selected as optimal, whether we posit voiceless coda or voiced coda as the input.

(11) a. Singular [pɛt]

input: /pɛt/	*Voice	Ident-IO(voi)
☞ pɛt		
pɛd	*!	*

input: /pɛd/	*Voice	Ident-IO(voi)
☞ pɛt		*
pɛd	*!	

b. Singular [bɛt][2]

input: /bɛt/	*Voice	Ident-IO(voi)
☞ bɛt		
bɛd	*!	*

input: /bɛd/	*Voice	Ident-IO(voi)
☞ bɛt		*
bɛd	*!	

Note that these are the cases where multiple inputs converge on a single output form. Thus, we have to determine the real inputs. For this purpose, we call on LO, and the input forms which are most similar to the outputs are selected as the real inputs.

(12) Lexicon Optimization

a. Singular [pɛt]

inputs		outputs	*Voice	Ident-IO(voi)
☞ /pɛt/	→	[pɛt]		
/pɛd/	→	[pɛt]		*!

b. Singular [bɛt]

inputs		outputs	*Voice	Ident-IO(voi)
☞ /bɛt/	→	[bɛt]		
/bɛd/	→	[bɛt]		*!

[2] To avoid unnecessary complication, *Voice here is applied to the consonants in the relevant positions (i.e. coda and intervocalic positions) only. Thus, only one mark of violation is recorded for [bɛd], and none for [bɛt].

(13) Singular [pɛt], [bɛt]: contrast neutralized

inputs → output → real inputs

/pɛt/ → [pɛt] — LO → /pɛt/
/pɛd/ → [pɛt]

/bɛt/ → [bɛt] — LO → /bɛt/
/bɛd/ → [bɛt]

Unlike the singular examples [pɛt] and [bɛt] where all coda obstruents are neutralized into voiceless, the medial obstruents in [pɛ.tən] and [bɛ.dən] maintain the voicing contrast. As we pointed out in the previous section, contrasts are expressed in the outputs via faithfulness being ranked higher than markedness, i.e. IdentOns(voice) ≫ *Voice in the case at hand[3].

(14) a. Plural [pɛ.tən]

input: /pɛt-ən/	IdentOns(voi)	*Voice
☞ pɛ.tən		
pɛ.dən	*!	*

b. Plural [bɛ.dən]

input: /bɛd-ən/	IdentOns(voi)	*Voice
bɛ.tən	*!	
☞ bɛ.dən		*

Since faithfulness outranks markedness, the output forms are identical to the input forms, and LO is not relevant here. The inputs are simply those that are identical to their respective output forms.

(15) Plural [pɛtən], [bɛdən]: contrast expressed

inputs → output → real inputs

/pɛt-ən/ → [pɛt-ən] → /pɛt-ən/

/bɛd-ən/ → [bɛd-ən] → /bɛd-ən/

---

[3] Another constraint that might come into play here is Inter-V-Voice, which ensures that intervocalic consonants are voiced. If this constraint should be introduced, it must be dominated by IdentOns(voice) (but its ranking with respect to *Voice is not determined).

There is a potential problem involved here. LO selects as inputs /pɛt, pɛtən/ for [pɛt]~[pɛtən], and /bɛt, bɛdən/ for [bɛt]~[bɛ.dən]. Of notice here is the fact that the latter pair of underlying representations, /bɛt, bɛd-ən/, does not share a common input form. This challenges the standard view of UR that recognizes UR as a voucher for "base-identity" in morphologically related paradigms. That is, the forms in the output paradigm, [bɛt]~[bɛ.dən], are related morphologically, and thus their inputs must share a morpheme. However, this is not the case here.

Note the input forms chosen by LO here are all identical to the output forms. It seems that the problem with LO in allomorphy cases arises because LO as formulated in (2) always selects the input form that is identical to an output form. In the cases where different inputs map to different outputs so that outputs do not converge (cases of contrast expressed), for example, LO is not relevant and inputs are simply identical to their output forms, as we have seen above. In the cases where multiple inputs converge on a single output form (cases of contrast neutralized), on the other hand, LO is active but still selects the input form that is most similar to the output. That is, although markedness constraints dominate faithfulness constraints here, all the inputs realize into the same output form and thus there is a tie on the higher ranked markedness constraints (either by violation or by satisfaction). Accordingly, the next highest ranked faithfulness constraints act as a determiner, and the input form most similar to the output is chosen as optimal by LO.

In summary, LO encounters a potential problem when it deals with an allomorphy case such as Dutch voicing alternation, because the underlying representations chosen by LO are all identical to their output forms and thus there arises a situation where words in morphologically related paradigm do not share a morpheme, contra the standard view of UR. In the next section, we will examine the proposals advanced to treat the problem with LO in dealing with the allomorphy cases.

4. Allomorphy with a Unique UR

We have indicated that the main problem with LO is that it always selects an input most similar to the output and hence is indeterminate when faced with alternation. In Dutch alternation, [bɛt]~[bɛ.dən], for example, LO chooses /bɛt/ if presented with [bɛt] only; but if presented with [bɛdən] only, it chooses /bɛd/ minimizing input-output disparities.

Maintaining the traditional view that allomorphy is expressed by a single UR, Tesar/Smolensky (1996, 2000) propose "Paradigm-level Lexicon Optimization," where optimization applies not to individual forms, but to the entire paradigms. In addition, this version of LO is equipped with another assumption that identity of the expression of a morpheme is required across its paradigm, which is characterized by an output-output constraint, Ident-OO(voice) (cf. Burzio 1996, 1997, 2000; Benua 1995; Kenstowicz 1996; McCarthy 1996). With these assumptions, Tesar/Smolensky show that Paradigm-level LO is performed as in (17).

(16) Ident-OO(voice)
Corresponding segments of the base and the affixed form must agree in voicing.

(17) Paradigm-level Lexicon Optimization

	inputs	outputs	IdentOns (voi)	*Voi	Ident-IO(voi)	Ident-OO(voi)
☞	/bɛd + ø / ən / →	[bɛt] [bɛ.d-ən]		*	*	*
	/bɛt + ø / ən / →	[bɛt] [bɛ.d-ən]	*!	*	*	*

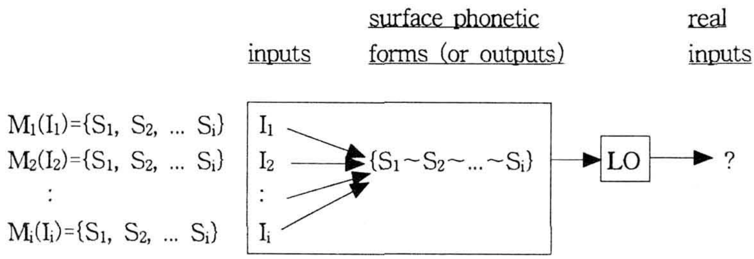
Here, LO compares two mapping relations to the output paradigm [bɛt]~[bɛd-ən], one from the input /bɛd/ and the other from /bɛt/, of which constraint violations are counted on the entire paradigms, rather than on individual forms. After the whole task of evaluation is completed, /bɛd/ is chosen as the real input form for the output paradigm [bɛt]~[bɛd-ən].

Although they have not stated explicitly, Tesar/Smolensky's Paradigm-level Lexicon Optimization is essentially identical to the version of LO proposed by Inkelas (1995), Alternation-Sensitive Lexicon Optimization.

(18) Alternation-sensitive restatement of Lexicon Optimization
"a grammar G and a set of $S=\{S_1, S_2, ..., S_i\}$ of surface phonetic forms for a morpheme M, suppose that there is a set of inputs $I=\{I_1, I_2, ..., I_i\}$, each of whose members has a set of surface realizations equivalent to S. There is some $I_i \in I$ such that the mapping between I_i and the members of S is the most harmonic with respect to G, i.e. incurs the fewest marks for the highest ranked constraints. The learner should choose I_i as the underlying representation for M."

According to this version of LO, mapping to the entire allomorphy paradigm, rather than to a single output form, is fundamental to the computation of a real input form. This is exactly what Prince/Smolensky's Paradigm-level LO aims at. The following diagram, which schematizes the situation depicted in (18) where multiple inputs converge on an alternation paradigm, further demonstrates that LO performs on the mapping relations, $M_1(I_1)=\{S_1, S_2, \dots S_i\}$, $M_2(I_2)=\{S_1, S_2, \dots S_i\}$, ..., $M_i(I_i)=\{S_1, S_2, \dots S_i\}$, rather than on individual surface forms.

(19)



We should note here that for LO to be active, the output paradigm {S1~S2~...~Si} must be optimal in each mapping. In Dutch voicing alternation, therefore, LO is called upon only if the paradigm {[bɛt]~[bɛd]} is chosen as the optimal output without respect to its input shapes.

Tesar/Smolensky's Paradigm-level LO of Dutch voicing alternation (17) encounters a problem in this respect. As we will see below, Paradigm-level LO is performed even though the inputs do not converge on the output paradigm, [bɛt]~[bɛ.dən]. The input /bɛd/ maps to [bɛt]~[bɛ.dən], while the input /bɛt/ to *[bɛt]~[bɛ.tən].

(20) Constraint tableaux for [bɛt]~[bɛ.d-ən]

a. Base input: /bɛd/

/bɛd + ø/ən /	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
☞ [bɛt] [bɛ.dən]		 *	* 	*
[bɛd] [bɛ.dən]		*! *		
[bɛd] [bɛ.tən]	 *!	* 	 *	*
[bɛt] [bɛ.tən]	*!		* *	

b. Base input: /bɛt/

/bɛt + ø/ən /	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
[bɛt] [bɛ.dən]	*!	*	*	*
[bɛd] [bɛ.dən]	*!	* *	* *	
[bɛd] [bɛ.tən]		*!	*	*
💣 [bɛt] [bɛ.tən]				

The situation does not change under different constraint systems. If we posit new markedness constraints *VoiceCoda (voiced coda obstruents are prohibited) and Inter-V-Voice (intervocalic consonants are voiced), and place *VoiceCoda on the top of the constraint ranking, the inputs still do not produce a unique paradigm. Here, too, LO does not even have a chance to apply and /bɛd/, from which the optimal output paradigm is produced, is determined as the input form[4].

(21) [bɛt]~[bɛ.dən] under a different constraint ranking

/bɛd + ø/ən /	*VoiCoda	Ident-IO (voi)	Inter-V-Voice	Ident-OO (voi)
☞ [bɛt] [bɛ.dən]		*		*
[bɛd] [bɛ.dən]	*!			
[bɛd] [bɛ.tən]	*!	*	*	*
[bɛt] [bɛ.tən]		* *!	*	

[4] It appears that the output paradigm [bɛt]~[bɛdən] is obtained regardless of its input, if the ranking between Ident-IO(voice) and Inter-V-Voice is reversed so that the overall ranking is *VoiceCoda ≫ Inter-V-Voice ≫ Ident-IO(voi) ≫ Ident-OO(voi). This ranking, however, does not produce the other set of output forms, [pɛt] ~[pɛtən], where obstruent voicing does not show alternation.

/pɛt + ø/ən /	*VoiCoda	Inter-V-Voice	Ident-IO (voi)	Ident-OO (voi)
💣 [pɛt] [pɛ.dən]			*	*
[pɛd] [pɛ.dən]	*!		* *	
[pɛd] [pɛ.tən]	*!	*	*	*
[pɛt] [pɛ.tən]		*!		

/bɛt + ø ən /	*VoiCoda	Ident-IO (voi)	Inter-V-Voice	Ident-OO (voi)
[bɛt] [bɛ.dən]		*!		*
[bɛd] [bɛ.dən]	*!	* *		
[bɛd] [bɛ.tən]	*!	*	*	*
☞ [bɛt] [bɛ.tən]			*	

Now we see that in any cases we have examined above, multiple inputs map separately to their own output paradigms, not converging on a uniform output paradigm. Hence, LO does not apply, whether it is the version of LO proposed by Tesar/Smolensky or by Inkelas, and a particular input must be set for an output paradigm. This goes against the fundamental spirit of ROB, which imposes no restriction whatsoever on underlying representations.

In summary, following the long-standing traditional view that morphological relation is reflected in the UR, Tesar/Smolensky and Inkelas both deal with output allomorphs as a paradigm, rather than individual forms. In this way, they contend, they can make sure that words in a morphologically related paradigm share a morpheme, doing away with the problem that arises when LO is applied in alternating cases. As we have seen in the treatment of Dutch voicing alternation, however, different input forms do not converge on a single output paradigm. Thus, LO is not invoked and a particular form is determined as the input, contra ROB.

5. Allomorphy without a Unique UR

We have seen that Tesar/Smolensky (1996, 2000) and Inkelas (1995) tackle the issue based on the idea that words in a morphologically related paradigm must share a unique UR. Kager (1999), however, notes that there is an overlap in functions between UR and base-identity, both of which maximize uniform exponence (uniform phonological spellout of morphology), and that the standard model of OT relies on output-output correspondence to treat the notion of uniform exponence (Benua 1995, McCarthy 1996, Kenstowicz 1996, Steriade 1996). Thus, the standard theory of OT is conceptually redundant because it has two means, UR and output-output correspondence, to achieve a single goal, uniform exponence (cf. Burzio 1996, 1997, 2000).

To effectively deal with the problem, (Kager 1999 : 415) discards the notion of a unique UR for allomorphy in favor of output-output correspondence, and proposes a model in which "[t]he lexicon no longer supplies a unique UR for each morpheme, but instead it supplies a set of shape variants of the morpheme, allomorphs, chunks ready for insertion in various morphological contexts (base or affixed forms)."

(22) Allomorphic model without UR

	OO-Identity	
Output	Base ⇔ Affixed form	
	⇕ ⇕	*IO-Faithfulness*
Input	{Base allomorph, Contextual allomorph}	

In this model, inputs and outputs map with each other via paradigms, rather than individual forms, and accordingly, harmonic evaluation is also performed on paradigms. Kager shows that an output paradigm in Dutch, [bɛt]~[bɛd-ən], is produced from an input paradigm, {bɛt~bɛd-ən} or {bɛd~bɛd-ən}, and in a similar fashion, [pɛt]~[pɛt-ən] is obtained from an input paradigm, {pɛt~pɛt-ən} or {pɛd~pɛt-ən}.

(23)

<u>inputs</u>		<u>outputs</u>	<u>inputs</u>		<u>outputs</u>
{bɛt~bɛd-ən}	→	[bɛt]~[bɛ.d-ən]	{pɛt~pɛt-ən}	→	[pɛt]~[pɛ.t-ən]
{bɛd~bɛd-ən}	↗		{pɛd~pɛt-ən}	↗	

(24) Constraint tableaux for Dutch

a. [bɛt]~[bɛd-ən]

Input: {bɛt~bɛd-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*		*
bɛt~bɛ.t-ən	*!		*	
bɛd~bɛ.d-ən		**!	*	
bɛd~bɛ.t-ən	*!	*	**	*

Input: {bɛd~bɛd-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*	*	*
bɛt~bɛ.t-ən	*!		**	
bɛd~bɛ.d-ən		**!		
bɛd~bɛ.t-ən	*!	*	*	*

b. [pɛt]~[pɛt-ən]

Input: {pɛt~pɛt-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
pɛt~pɛ.d-ən	*!	*	*	*
☞ pɛt~pɛ.t-ən				
pɛd~pɛ.d-ən	*!	**	**	
pɛd~pɛ.t-ən		*!	*	*

Input: {pɛd~pɛt-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
pɛt~pɛ.d-ən	*!	*	**	*
☞ pɛt~pɛ.t-ən			*	
pɛd~pɛ.d-ən	*!	**	*	
pɛd~pɛ.t-ən		*!		*

Unlike the versions of LO proposed by Tesar/Smolensky (1996, 2000) and Inkelas (1995), Kager's model brings about a situation where multiple inputs *do* converge on a single output. Hence, LO is invoked to choose {bɛt~bɛd-ən} and {pɛt~pɛt-ən} as the real input paradigms.

(25) Lexicon Optimization

Inputs outputs	IDOns (voi)	*Voi	ID-IO (voi)	ID-OO (voi)
☞ {bɛt~bɛd-ən} → bɛt~bɛ.d-ən		*		*
{bɛd~bɛd-ən} → bɛt~bɛ.d-ən		*	*!	*

Inputs outputs	IDOns (voi)	*Voi	ID-IO (voi)	ID-OO (voi)
☞ {pɛt~pɛt-ən} → pɛt~pɛ.t-ən				
{pɛd~pɛt-ən} → pɛt~pɛ.t-ən			*!	

We have yet to address another important issue: there are other plausible input paradigms, {bɛd~bɛt-ən} & {bɛt~bɛt-ən} (for [bɛt]~[bɛd-ən]) and {pɛt ~pɛd-ən} & {pɛd~pɛd-ən} (for [pɛt]~[pɛt-ən]), but they have been excluded from consideration so far. The reason is that if they are counted in, they would produce incorrect *[bɛt]~[bɛt-ən] and *[pɛt]~[pɛ.d-ən] as the optimal outputs.

(26) a. Inputs: {bεd~bεt-ən} and {bεt~bεt-ən}

Input: {bεd~bεt-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
bεt~bε.d-ən	*!	*	**	*
☞ bεt~bε.t-ən			*	
bεd~bε.d-ən	*!	**	*	
bεd~bε.t-ən		*!		*

Input: {bεt~bεt-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
bεt~bε.d-ən	*!	*	*	*
☞ bεt~bε.t-ən				
bεd~bε.d-ən	*!	**	**	
bεd~bε.t-ən		*!	*	*

b. Inputs: {pεt~pεd-ən} and {pεd~pεd-ən}

Input: {pεt~pεd-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
☞ pεt~pε.d-ən		*		*
pεt~pε.t-ən	*!		*	
pεd~pε.d-ən		**!	*	
pεd~pε.t-ən	*!	*	**	*

Input: {pεd~pεd-ən}	IdentOns (voi)	*Voi	Ident-IO (voi)	Ident-OO (voi)
☞ pεt~pε.d-ən		*	*	*
pεt~pε.t-ən	*!		**	
pεd~pε.d-ən		**!		
pεd~pε.t-ən	*!	*	*	*

The question then is how we formally restrict input paradigms to {bεt~bεd-ən} & {bεd~bεd-ən}, on the one hand, and {pεt~pεt-ən} & {pεd~pεt-ən}, on the other, as in (24). As we have seen, the voicing contrast in Dutch is expressed in intervocalic positions. The input paradigms that we have considered for evaluation in (24) are those in which the forms in intervocalic positions are invariant from input to output. The input paradigms that must be ruled out, on the other hand, include the forms in which the intervocalic obstruents are different from input to output. It seems, then, the input form is always identical to its output in the positions where contrast is preserved. Capitalizing on this observation, I propose the following meta-principle of the grammar that limits the input paradigms subjected to GEN.

(27) Input-Output Isomorphy (IOI)
Where contrast is expressed, the input is isomorphous to its output.

The presence of IOI was hinted earlier in this paper. In section 2, we already made it clear that in the cases where contrast are expressed, faithfulness outranks markedness, giving rise to input-output identity. We demonstrated that in those cases, the forms that are identical to the outputs are simply determined as the input forms. The following diagram schematizes how IOI, along with LO, applies to the Dutch case.

(28)

IOI

{bɛt~bɛd-ən}
{bɛt~bɛt-ən}
{bɛd~bɛd-ən}
{bɛd~bɛt-ən}

inputs: {bɛt~bɛd-ən} {bɛd~bɛd-ən} → output: [bɛt]~[bɛd-ən] — LO → real input: {bɛt~bɛd-ən}

In summary, Kager's mode of LO assumes that the lexicon supplies a set of allomorphs, rather than a unique UR for each morpheme. We have seen that it fares best in treating allomorphy cases, if it is corroborated with IOI, which guarantees inputs to be identical to the outputs in the cases where contrasts are expressed. In the next section, Kager's model will be further tested on the typological variants of Dutch.

6. LO and the Typological Variants of Dutch

We have seen that the voicing contrast in Dutch is neutralized (CN) in word-final positions but is expressed (CE) in intervocalic positions. The typological variants of Dutch that we will examine are those that vary in the realization of voicing contrast: (i) Dutch′, where contrast is expressed word-finally and neutralized intervocalically; (ii) Dutch″, where contrast is neutralized both word-finally and intervocalically; and (iii) Dutch‴, where contrast is expressed both word-finally and intervocalically. The constraint ranking for each variant is given below in (30).

(29) Typological variants of Dutch

a.	Dutch	CN	[bɛt] [pɛt]	~ ~	[bɛdən] [pɛtən]	CE
b.	Dutch′	CE	[bɛd] [pɛt]	~ ~	[bɛdən] [pɛdən]	CN
c.	Dutch″	CN	[bɛt] [pɛt]	~ ~	[bɛdən] [pɛdən]	CN
d.	Dutch‴	CE	[bɛd] [pɛt]	~ ~	[bɛdən] [pɛtən]	CE

(30) Constrain ranking for the typological variant of Dutch[5]

a. Dutch: IdentOns(voi) ≫ *Voice, IVV ≫ Ident-IO(voi)
b. Dutch′: IVV ≫ IdentOns(voi) ≫ Ident-IO(voi) ≫ *Voice
c. Dutch″: IVV ≫ *Voice ≫ IdentOns(voi) ≫ Ident-IO(voi)
d. Dutch‴: IdentOns(voi) ≫ Ident-IO(voi) ≫ *Voice, IVV

Let us briefly discuss how the constraint ranking for each variant of Dutch is determined. First, in Dutch′, voicing contrast is expressed in word-final positions, and thus Ident-IO(voice) outranks *Voice. Voicing contrast is neutralized in intervocalic positions, which suggests that Inter-V-Voice dominates faithfulness constraints, IdentOns(voice) and Ident-IO(voice). Panini's Theorem determines the ranking between the two faithfulness constraints such that a specific constraint, IdentOns(voice), dominates a general constraint, Ident-IO(voice).

Second, the neutralization of voicing contrast in Dutch″ occurs in word-final positions as well as in intervocalic positions, suggesting that markedness constraints, *Voice and Inter-V-Voice, dominate faithfulness constraints. Of the two markedness constraints, Inter-V-Voice outranks *Voice because voiced consonants in intervocalic positions ([pɛ.dən] and [bɛ.dən]) violate the latter but observe the former. Finally, voicing contrast in Dutch‴ is expressed both in word-final positions and in intervocalic positions, due to faithfulness constraints outranking markedness constraints.

[5] It appears that Inter-V-Voice, which we have not fully made use of in discussing Dutch, is important when we compare the typological variants of Dutch. This constraint, if introduced for Dutch, is dominated by IdentOns(voice) as we have seen in footnote 3. The introduction of Inter-V-Voice does not affect the conclusion we drew in the previous section.

Kager's Allomorphy Model, with the role of IOI fully recognized, provides a good account of each typological variant of Dutch. First, in Dutch′, the voicing contrast is expressed in word-final positions, and thus IOI sets the inputs for [bεd] and [pεt] as /bεd/ and /pεt/, respectively. Thus, the input paradigms that we consider for Dutch′ are restricted to {bεd~bεt-ən} & {bεd~bεd-ən}, and {pεt~pεt-ən} & {pεt~pεd-ən}. These input paradigms, after they go through the constraint system, produce the output paradigms, [bεd]~[bε.d-ən] and [pεt]~[pε.d-ən], respectively.

(31)

inputs		outputs	inputs		outputs
{bεd~bεt-ən}	→	[bεd]~[bε.d-ən]	{pεt~pεt-ən}	→	[pεt]~[pε.d-ən]
{bεd~bεd-ən}	↗		{pεt~pεd-ən}	↗	

(32) Dutch′ (contrast is expressed word-finally, but neutralized intervocalically)

a. output: [bεd]~[bε.d-ən]

Input: {bεd~bεt-ən}	IVV	IdentOns (voi)	Ident-IO (voi)	*Voi	Ident-OO (voi)
bεt~bε.d-ən		*!	**	*	*
bεt~bε.t-ən	*!		*		
☞ bεd~bε.d-ən		*	*	**	
bεd~bε.t-ən	*!			*	*

Input: {bεd~bεd-ən}	IVV	IdentOns (voi)	Ident-IO (voi)	*Voi	Ident-OO (voi)
bεt~bε.d-ən			*!	*	*
bεt~bε.t-ən	*!	*	**		
☞ bεd~bε.d-ən				**	
bεd~bε.t-ən	*!	*	*	*	*

b. output: [pεt]~[pε.d-ən]

Input: {pεt~pεt-ən}	IVV	IdentOns (voi)	Ident-IO (voi)	*Voi	Ident-OO (voi)
☞ pεt~pε.d-ən		*	*	*	*
pεt~pε.t-ən	*!				
pεd~pε.d-ən		*	**!	**	
pεd~pε.t-ən	*!		*	*	*

Input: {pεt~pεd-ən}	IVV	IdentOns (voi)	Ident-IO (voi)	*Voi	Ident-OO (voi)
☞ pεt~pε.d-ən				*	*
pεt~pε.t-ən	*!	*	*		
pεd~pε.d-ən			*!	**	
pεd~pε.t-ən	*!	*	**	*	*

Here we have a case where multiple inputs converge on an output. LO applies now to choose {bɛd~bɛd-ən} and {pɛt~pɛd-ən} as the inputs.

(33) Lexicon Optimization: Dutch′

Inputs outputs	IVV	IDOns (voi)	ID-IO (voi)	*Voi	ID-OO (voi)
{bɛd~bɛt-ən} → bɛd~bɛ.d-ən		*!	*	**	
☞ {bɛd~bɛd-ən} → bɛd~bɛ.d-ən				**	

Inputs outputs	IVV	IDOns (voi)	ID-IO (voi)	*Voi	ID-OO (voi)
{pɛt~pɛt-ən} → pɛt~pɛ.d-ən		*!	*	*	*
☞ {pɛt~pɛd-ən} → pɛt~pɛ.d-ən				*	*

Second, in Dutch″, voicing contrasts are neutralized in word-final and intervocalic positions, yielding the output patterns, [bɛt]~[bɛ.d-ən] and [pɛt]~[pɛ.d-ən]. Here, there are at least two potential inputs for each position, and hence, logically, there are at least four input paradigms to consider for each output paradigm. These input paradigms all converge on an output paradigm, and accordingly LO is invoked here to choose {bɛt~bɛd-ən} and {pɛt~pɛd-ən} as the real input paradigm[6].

(34) <u>inputs</u> <u>outputs</u> <u>inputs</u> <u>outputs</u>

{bɛt~bɛt-ən} {pɛt~pɛt-ən}
{bɛt~bɛd-ən} → [bɛt]~[bɛ.d-ən] {pɛt~pɛd-ən} → [pɛt]~[pɛ.d-ən]
{bɛd~bɛt-ən} {pɛd~pɛt-ən}
{bɛd~bɛd-ən} {pɛd~pɛd-ən}

(35) Dutch″ (contrast is neutralized both word-finally and intervocalically)
output: [bɛt]~[bɛ.d-ən]

Input: {bɛt~bɛt-ən}	IVV	*Voi	IdentOns (voi)	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*	*	*	*
bɛt~bɛ.t-ən	*!				
bɛd~bɛ.d-ən		**!	*	**	
bɛd~bɛ.t-ən	*!	*		*	*

6) For space limitation, we will only consider [bɛt]~[bɛ.d-ən]. The constraint tableaux for [pɛt]~[pɛ.d-ən] is exactly the same as those for [bɛt]~[bɛ.d-ən]. Further, LO for [pɛt]~[pɛ.d-ən] applies in the same mode as LO for [bɛt]~[bɛ.d-ən] does, and {pɛt~pɛ.d-ən} is selected as the optimal input paradigm.

Input: {bɛt~bɛd-ən}	IVV	*Voi	IdentOns (voi)	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*			*
bɛt~bɛ.t-ən	*!		*	*	
bɛd~bɛ.d-ən		**!		*	
bɛd~bɛ.t-ən	*!	*	*	**	*

Input: {bɛd~bɛt-ən}	IVV	*Voi	IdentOns (voi)	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*	*	**	*
bɛt~bɛ.t-ən	*!			*	
bɛd~bɛ.d-ən		**!	*	*	
bɛd~bɛ.t-ən	*!	*			*

Input: {bɛd~bɛd-ən}	IVV	*Voi	IdentOns (voi)	Ident-IO (voi)	Ident-OO (voi)
☞ bɛt~bɛ.d-ən		*		*	*
bɛt~bɛ.t-ən	*!		*	**	
bɛd~bɛ.d-ən		**!			
bɛd~bɛ.t-ən	*!	*	*	*	*

(36) Lexicon Optimization: Dutch''

Inputs outputs	IVV	*Voi	IDOns (voi)	ID-IO (voi)	ID-OO (voi)
{bɛt~bɛt-ən} → bɛt~bɛ.d-ən		*	*!	*	*
☞ {bɛt~bɛd-ən} → bɛt~bɛ.d-ən		*			*
{bɛd~bɛt-ən} → bɛt~bɛ.d-ən		*	*!	**	*
{bɛd~bɛd-ən} → bɛt~bɛ.d-ən		*		*!	*

Finally, in Dutch''', voicing contrast is expressed both in word-final and intervocalic positions. Here, IOI delimits possible inputs to those that are identical to the output paradigms.

(37) Dutch''' (contrast is expressed word-finally and intervocalically)

a. output: [bɛd]~[bɛ.d-ən]

Input: {bɛd~bɛd-ən}	IdentOns (voi)	Ident-IO (voi)	*Voi	IVV	Ident-OO (voi)
bɛt~bɛ.d-ən		*!	*		*
bɛt~bɛ.t-ən	*!	**		*	
☞ bɛd~bɛ.d-ən			**		
bɛd~bɛ.t-ən	*!	*	*	*	*

b. output: [pɛt]~[pɛ.t-ən]

Input: {pɛt~pɛt-ən}	IdentOns (voi)	Ident-IO (voi)	*Voi	IVV	Ident-OO (voi)
pɛt~pɛ.d-ən	*!	*	*		*
☞ pɛt~pɛ.t-ən				*	
pɛd~pɛ.d-ən	*!	**	**		
pɛd~pɛ.t-ən		*!	*!	*	*

In summary, we have seen that Kager's Allomorphy Model, geared with IOI that effectively delimits the inputs subjected to GEN, deals successfully with voicing alternation in each typological variant of Dutch.

7. Conclusion

Centering on LO, we have so far discussed how we should treat UR in OT. First, we examined Prince/Smolensky's (1993) version of LO. Their proposal is based on the assumptions that a unique UR is posited for allomorphs and that LO applies to individual forms. We have noted that this model encounters a contradiction, since after LO applies, allomorphs are not represented with a unique UR.

We then discussed Tesar/Smolensky's (1996, 2000) and Inkelas' (1995) approach to LO. Like Prince/Smolensky, they both begin with an assumption that allomorphs are characterized by a unique UR. Unlike Prince/Smolensky, however, they propose that LO is performed at the level of paradigms, rather than individual forms. Based on the Dutch data of voicing alternation, we have shown that different inputs do not converge on a single output in this model, and hence LO is not invoked so that a particular form is fixed as the input for an output, contra ROB.

Finally, we have examined Kager's (1999) model of LO. Kager deviates from Prince/Smolensky, and discards both of their assumptions. In particular, he poses a paradigm, rather than an individual form, as the basic unit of computation. Further, he discards the notion that morphologically related words share a unique UR, since such a notion is already covered by output-output correspondence. We have shown that if this model is properly fortified by IOI, which guarantees an input to be identical to the output in the cases where contrasts are expressed, it provides a good account of voicing alternation not only in Dutch but also in its typological variants.

References

Benua, L. (1995) 'Identity Effects in Morphological Truncation,' in J. Beckman, L. W. Dickey, & S. Urbanczyk, eds., *University of Massachusetts Occasional Papers in Linguistics* 18: *Papers in Optimality Theory*, 77-136, Graduate Linguistic Student Association, Amherst, Massachusetts.

Burzio, L. (1996) 'Surface Constraints vs. Underlying Representations,' in J. Durand & B. Laks, eds., *Current Trends in Phonology: Models and Methods*, 97-122, University of Salford Publications.

Burzio, L. (1997) 'Strength in Numbers,' in V. Miglio and B. Moren, eds., *University of Maryland Working Papers in Linguistics 5*, 27-52.

Burzio, L. (2000) 'Cycles, Non-Derived-Environment-Blocking, and Correspondence,' in J. Dekkers, F. van der Leeuw and J. van der Weijer, eds., *Optimality Theory: Syntax, Phonology, and Acquisition*, 47-87, Oxford University Press.

Hong, Sunghoon (2001) 'Richness of the Base, Lexicon Optimization, and Suffix /ɨ/~ø Alternation in Korean,' to appear in *Studies in Phonetics, Phonology and Morphology* 7.

Inkelas, S. (1995) 'The Consequences of Optimization for Underspecification,' in J. Beckman, ed., *Proceedings of the North East Linguistic Society* 25, 287-302.

Kager, R. (1999) *Optimality Theory*, Cambridge University Press.

Kenstowicz, M. (1996) 'Base-Identity and Uniform Exponence: Alternatives to Cyclicity,' in J. Durand & B. Laks, eds., *Current Trends in Phonology: Models and Methods*, 363-393, University of Salford Publications.

Lombardi, L. (1999) 'Positional Faithfulness and Voicing Assimilation in Optimality Theory,' *Natural Language and Linguistic Theory* 17, 267-302.

McCarthy, J. (1996) 'Extensions of Faithfulness: Rotuman Revisited,' ms., University of Massachusetts, Amherst.

Prince, A. and P. Smolensky (1993) *Optimality Theory: Constraint Interaction in Generative Grammar*, MIT Press, TR-2, Rutgers University Cognitive Science Center.

Smolensky, P. (1996) 'The Initial State and Richness of the Base in Optimality Theory,' ms., Johns Hopkins University.

Steriade, D. (1996) 'Paradigm Uniformity and the Phonetics-Phonology

Boundary,' ms., UCLA.
Tesar, B. and P. Smolensky (1996) 'Learnability in Optimality Theory (long version),' Technical Report JHU-CogSci-96-4, Department of Cognitive Science, Johns Hopkins University.
Tesar, B. and P. Smolensky (1998) 'Learnability in Optimality Theory,' *Linguistic Inquiry* 29, 229-268.
Tesar, B. and P. Smolensky (2000) *Learnability in Optimality Theory*, the MIT Press, Cambridge, Massachusetts.

Department of English Language and Literature
Hansung University
389 Samseon-dong 2-ga, Seongbuk-gu
Seoul 136-792, Korea
E-mail : hongsh@hansung.ac.kr